



Data Acquisition System of the Virgo Gravitational Waves Interferometric Detector

F. Acernese, P. Amico, M. Alshourbagy, F. Antonucci, S. Aoudia, P. Astone, S. Avino, D. Babusci, G. Ballardin, F. Barone, et al.

► To cite this version:

F. Acernese, P. Amico, M. Alshourbagy, F. Antonucci, S. Aoudia, et al.. Data Acquisition System of the Virgo Gravitational Waves Interferometric Detector. 11th IEEE International Symposium on Distributed Simulation and Real Time Applications, Oct 2007, Chania, Greece. in2p3-00191035

HAL Id: in2p3-00191035

<https://hal.in2p3.fr/in2p3-00191035>

Submitted on 23 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Data Acquisition System of the Virgo Gravitational Waves Interferometric Detector

F. Acernese⁶, P. Amico¹⁰, M. Alshourbagy¹¹, F. Antonucci¹², S. Aoudia⁷, P. Astone¹², S. Avino⁶, D. Babusci⁴, G. Ballardin², F. Barone⁶, L. Barsotti¹¹, M. Barsuglia⁸, F. Beauville¹, S. Bigotta¹¹, S. Birindelli¹¹, M.A. Bizouard⁸, C. Boccara⁹, F. Bondu⁷, L. Bosi¹⁰, C. Bradaschia¹¹, S. Braccini¹¹, A. Brillet⁷, V. Brisson⁸, L. Brocco¹², D. Buskulic¹, E. Calloni⁶, E. Campagna³, F. Carbognani², F. Cavalier⁸, R. Cavalieri², G. Cella¹¹, E. Cesarini³, E. Chassande-Mottin⁷, N. Christensen², C. Corda¹¹, A. Corsi¹², F. Cottone¹⁰, A.-C. Clapson⁸, F. Cleva⁷, J.-P. Coulon⁷, E. Cuoco², A. Dari¹⁰, V. Dattilo², M. Davier⁸, M. del Prete², R. De Rosa⁶, L. Di Fiore⁶, A. Di Virgilio¹¹, B. Dujardin⁷, A. Eleuteri⁶, I. Ferrante¹¹, F. Fidecaro¹¹, I. Fiori¹¹, R. Flaminio^{1,2}, J.-D. Fournier⁷, S. Frasca¹², F. Frasconi^{2,11}, L. Gammaitoni¹⁰, F. Garufi⁶, E. Genin², A. Gennai¹¹, A. Giazotto¹¹, G. Giordano⁴, L. Giordano⁶, R. Gouaty¹, D. Grosjean¹, G. Guidi³, S. Hebri², H. Heitmann⁷, P. Hello⁸, S. Karkar¹, S. Kreckelbergh⁸, P. La Penna², M. Laval⁷, N. Leroy⁸, N. Letendre¹, B. Lopez², Lorenzini³, V. Lorette⁹, G. Losurdo³, J.-M. Mackowski⁵, E. Majorana¹², C. N. Man⁷, M. Mantovani¹¹, F. Marchesoni¹⁰, F. Marion¹, J. Marque², F. Martelli³, A. Masserot¹, M. Mazzoni³, L. Milano⁶, F. Menzinger², C. Moins², J. Moreau⁹, N. Morgado⁵, B. Mours¹, F. Nocera², C. Palomba¹², F. Paoletti^{2,11}, S. Pardi⁶, A. Pasqualetti², R. Passaquieti¹¹, D. Passuello¹¹, B. Perniola³, F. Piergiovanni³, L. Pinard⁵, R. Poggiani¹¹, M. Punturo¹⁰, P. Puppo¹², K. Qipiani⁶, P. Rapagnani¹², V. Reita⁹, A. Remillieux⁵, F. Ricci¹², I. Ricciardi⁶, P. Ruggi², G. Russo⁶, S. Solimeno⁶, A. Spallicci⁷, R. Stanga³, M. Tarallo¹¹, M. Tonelli¹¹, A. Toncelli¹¹, E. Tournefier¹, F. Travasso¹⁰, C. Tremola¹¹, G. Vajente¹¹, D. Verkindt¹, F. Vetrano³, A. Viceré³, J.-Y. Vinet⁷, H. Vocca¹⁰ and M. Yvert¹

¹Laboratoire d'Annecy-le-Vieux de physique des particules (LAPP), IN2P3/CNRS, Université de Savoie, BP 110, F-74941, Annecy-le-Vieux, CEDEX, France;

²European Gravitational Observatory (EGO), Via E. Amaldi, I-56021 Cascina (PI) Italia;

³INFN, Sezione Firenze/Urbino, Via G.Sansone 1, I-50019 Sesto Fiorentino; and/or Università di Firenze, Largo E.Fermi 2, I - 50125 Firenze and/or Università di Urbino, Via S.Chiera, 27 I-61029 Urbino, Italia;

⁴INFN, Laboratori Nazionali di Frascati, Via E. Fermi, 40, I-00044 Frascati (Roma) - Italia;

⁵LMA, 22, Boulevard Niels Bohr, 69622 - Villeurbanne- Lyon Cedex, France;

⁶INFN, Sezione di Napoli and/or Università di Napoli "Federico II", Complesso Universitario di Monte S. Angelo Via Cintia, I-80126 Napoli, Italia and/or Università di Salerno, Via Ponte Don Melillo, I-84084 Fisciano (Salerno), Italia;

⁷Department Artemis - Observatoire de la Côte d'Azur, BP 42209, 06304 Nice Cedex 4, France;

⁸Laboratoire de l'Accélérateur Linéaire (LAL), IN2P3/CNRS-Université de Paris-Sud,

B.P. 34, 91898 Orsay Cedex - France;

⁹ESPCI - 10, rue Vauquelin, 75005 Paris - France;

¹⁰INFN, Sezione di Perugia and/or Università di Perugia, Via A. Pascoli, I-06123 Perugia - Italia;

¹¹INFN, Sezione di Pisa and/or Università di Pisa, Via Filippo Buonarroti, 2, I-56127 PISA - Italia;

¹²INFN, Sezione di Roma and/or Università "La Sapienza", P.le A. Moro, 2, I-00185, Roma.

Abstract—Virgo is an experiment aiming at the detection of gravitational waves emitted by astrophysical sources. Its detector, based on a 3km arms interferometer, is a complex setup which requires several digital control loops running up to 10kHz, an accurate and reliable central timing system and an efficient data acquisition, all of them being distributed over 3km. We overview here the main hardware and software components developed for the data acquisition system (DAQ) and its current architecture. Then, we briefly discuss its connections with interferometer's controls, especially through the automation of the interferometer's startup procedure. Then, we describe the tools used to

monitor the DAQ and the performances we measured with them. Finally, are described also the tools developed for the online detector monitoring, mandatory complement of the DAQ for the commissioning of the Virgo detector.

I. INTRODUCTION

The Virgo experiment [1][2] aims at the detection of gravitational waves of astrophysical origin by using a suspended Michelson interferometer with 3 km arms length. The commissioning of this interferometric detector reaches

currently its end and the sensitivity of the apparatus is approaching the aim requested for a possible gravitational wave detection, in a bandwidth spanning from a few Hertz to several kiloHertz.

As shown on Fig.1, Virgo detector is a Michelson interferometer using a 20 W Nd:YAG laser stabilized in power and frequency and an injection system that includes an input mode-cleaner (a 3 mirrors cavity maintained resonant on the laser TEM00 mode). The detection system includes a set of detection photodiodes and an output mode-cleaner that improves the dark fringe contrast. The main improvements with respect to a simple Michelson are 3 km long Fabry-Perot cavities in each arm that increase up to 100 km the optical arms length and a recycling mirror that increases the light power inside the interferometer. Once the interferometer is set on its working point (optical cavities resonant and interference kept on the dark fringe), the arms length modification due to a gravitational wave is detected by apparition of light at the interferometer's output.

To reduce as much as possible the environment noises that could affect the dark fringe signal, the main components of the interferometer are placed under vacuum and the mirrors are suspended under sophisticated anti-seismic suspensions that reduce the horizontal seismic motion by a factor 10^{14} above 10 Hz. The input and output mode-cleaner needs to be seismically isolated and are thus also suspended.

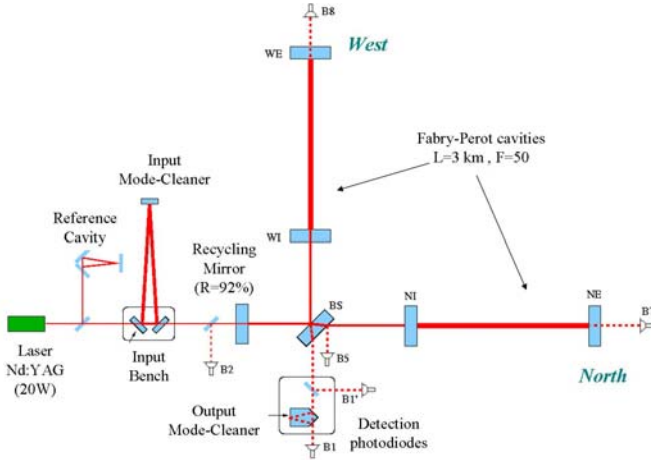


Fig. 1. Optical configuration of the Virgo interferometer..

Such a long base-line interferometer needs a data acquisition system [3][4] able to collect in a coherent way all the data needed for the signal extraction and the qualification of the detector. This requires a site wide timing system and several dedicated readouts interfaced with the control loops distributed over the site. It requires also tools to format, transport and write the data and online monitoring tools to display and analyse in real time the state of the detector. In detection configuration, the Virgo interferometer requires

many control loops, usually digital loops, that need to be developed and tuned [5][6]. Since most of the Virgo control systems are digital, the data acquisition (DAQ) has strong connections with them: on one hand, the timing system of the DAQ is the same as the one used to drive the controls, on the other hand, the digitized control loop signals are often collected synchronously by the DAQ. This put an additional strong constraint on the DAQ: to not disturb the digital controls. Given the complexity of the interferometer, the control systems have continuously evolved during the commissioning phase, with an increasing number of controls being turned on and a continuous noise hunting leading to a regular improvement of the detector sensitivity. This puts also flexibility and scalability constraints on the data acquisition. In addition, the DAQ has to challenge a strong request of reliability because the detector should be operated 24 hours per day, all year long.

We first overview, in section II, the standard and custom components of the DAQ chosen under those constraints. Then, section III describes the current architecture of the data acquisition, including online data access and online data processing set up with the DAQ software tools. Section IV is a short description of the two control setups where DAQ is strongly involved: the interferometer's control loops and the automation of the interferometer's startup procedure. Then, section V focus on the DAQ monitoring and on the DAQ performances measured. Finally, section VI deals with the main software tools developed for the online detector monitoring which is a mandatory complement of data acquisition.

II. STANDARD AND CUSTOM DAQ COMPONENTS

The choices made for the hardware and software components of the data acquisition have been guided by the constraints previously described and by the easiness of use and maintenance. Thus, whenever possible, standard solutions have been adopted for each DAQ hardware component. Moreover, based on those standard choices, a tool mandatory for the control of the interferometer and for the data acquisition has been developed: the timing system. Finally, a set of home-made libraries has been developed to provide plug-and-play DAQ software tools.

A. Standard components

For the front-end part, we use data transmission through optical links to reduce propagation delays and electromagnetic noise and VME numerical electronic boards associated with RIO8062 PowerPC CPU under LynXOS [7] to handle the real time of Virgo control loops and front-end data acquisition. In addition, home-made VME boards have been developed for the timing system [8], for the reception and transmission through optical links [9] and for the CCD cameras readout used in local controls and beam shape measurements [10].

For data collection and online processing, we use workstations under Linux SL4, shared memories and Gigabit Ethernet

with TCP/IP protocol to exchange the data between the processes. In addition, a set of home-made C libraries has been developed to easily propagate the large data flow of Virgo between DAQ processes.

B. The Virgo timing system

Control loops and data acquisition of a 3 km arms interferometer need to be synchronized over a large length scale. In addition, to guarantee a data analysis in coincidence with other interferometric detectors like LIGO, it is necessary to stamp the data with a common time reference.

To fulfill this requirement, a GPS based central timing system has been developed to synchronize the Virgo interferometer's controls and readouts. In addition, the GPS reference used to drive this timing system is cross-checked with a local Rubidium atomic clock (which is itself resynchronized periodically on GPS time). Details about the timing system can be found in [4].

Measurements have been done [11], within a Virgo-LIGO collaboration, to check the inter-experiment and end-to-end timing accuracy. First results are that LIGO and Virgo timing systems deliver a 1PPS signal with a jitter less than $1\mu s$ between each other and that the accuracy on the GPS time stamp difference between the two experiments is around 1 ms.

C. The DAQ libraries

The home-made libraries developed for timing and readout management have been described in [4]. Particular emphasis is given here to the home-made libraries developed for the data collection.

Once the data have been digitized and acquired synchronously thanks to front-end readouts paced by the timing system, they need to be formatted to be propagated on the Ethernet network and stored on disk. A data format for gravitational wave detectors has been developed since 1996 in the framework of a Virgo-LIGO collaboration [12]: the "frame" format. It has been adopted now by several other gravitational waves detectors all over the world. Two libraries, "Fr" and "Frv" [13][14], have been developed to manage the frame format and the vectors of data. All the DAQ and online processing make an extensive use of those two libraries. In addition, a Frame Distribution library "Fd" [15] has been developed to manage the input/output of the frame-formatted data. It is based on the use of a custom library (FdShm) [16] managing a shared memory and on the use of the "Cm" library [17] that encapsulates the TCP/IP Ethernet protocol. Thanks to "Fd", we have created "FdIOServer", elementary part of the DAQ shown on Fig. 2, that sends or receives frames through the Ethernet network, through a shared memory or through disk files.

In addition, "Cfgr", a general purpose library, has been developed to handle mainly the parsing of configuration files and the exchange of informations with the graphical interface that controls and monitors the DAQ processes.

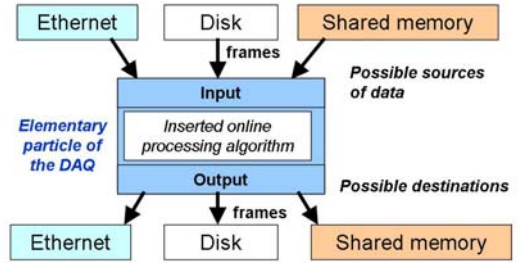


Fig. 2. Basic element of the data acquisition, created thanks to the "Fd" library. In this element can be introduced any online processing software developed in Virgo like for instance a resampling algorithm, a detector monitoring algorithm or a glitch search algorithm.

III. DATA ACQUISITION ARCHITECTURE

As shown on Fig.3, the Virgo data acquisition architecture can be split into three levels: the "front-end readouts", the "frame builders" and the "frame collectors". Those three levels have been mainly described in [4].

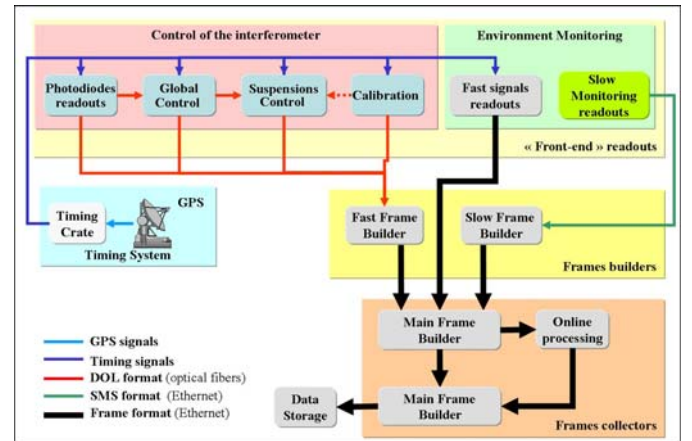


Fig. 3. General architecture of the Virgo data acquisition, with the 3 levels "front-end", "frame builders" and "frame collectors". Are also shown the GPS based timing system that rhythms the front-end readouts and the various data format used.

- **First level:** The front-end readouts run on RIO8062 CPU or on custom DSP. They are synchronized by the timing system and produce all the raw data acquired in Virgo. Some of them are also involved in the control loops and need to be reliable and fast enough to prevent any unlock. As a consequence, those processes do not build frames but send their data through optical links down to frame builders.
- **Second level:** The frame builders are a key feature of the data collection. They collect data from the various front-end readouts and put them under frame format. Two specific softwares have been developed in Virgo to this purpose: the "slow frame builder" [18] and the "fast frame builder". The fast frame builder run on RIO8062 CPU and uses a custom data compression algorithm to reduce

by about 2.5 the size of the frames sent through Ethernet down to the frames collectors.

- **Third level:** The frame collectors, historically called "main frame builders" [19], run on Linux workstations. They collect asynchronously the frames sent by several frames builders and merge them using the frame number. The result of the merging is a frame written in a shared memory [16] and immediately available online for next steps of the DAQ chain or online data access purposes.

As can be seen on Fig.4, the main modifications with respect to the architecture described in [4] is that the raw data collected are used for the automation of the interferometer's startup and also sent to several online processing tasks. In a last step, the results of the online processing are merged with the main data stream through a last frame collector (FbmSt) which sends the resulting frames down to data storage.

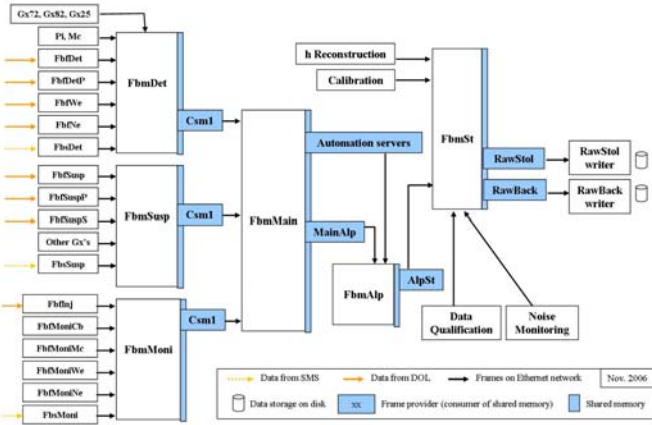


Fig. 4. Configuration of the data acquisition system in November 2006. Boxes at left aim at describing all the processes of the "front-end" and "frame builders" levels, that means about 110 processes distributed over 85 CPUs and DSPs. The data are collected by three upstream "frames collectors" FbmMoni, FbmSusp and FbmDet and passed to the automation system (FbmMain, Automation servers, FbmAlp). A last frames collector, FbmSt, adds to the main stream the data provided by the calibration system and by some of the online processing tasks (h Reconstruction, detector monitoring and noise monitoring). Data storage (RawStol) is securized by a parallel storage (RawBack).

A. Data streams and data access

Within this architecture, several data streams are produced. The main Virgo data stream is made of all the raw data collected through the three levels previously described. It represents today a total of about 2340 channels acquired (1370 fast channels and 970 slow monitoring channels), for a total data rate of about 20 MB/s (8 MB/s after the data compression done by the frame builders), which means about 690 GBytes per day written on disk [20].

Mainly, two other data streams are produced by some of the online processing tasks implemented during the Virgo commissioning: the 50 Hz resampling of the data and the trend data computation. As shown on Fig. 5, the 50 Hz resampling is distributed over the three upstream frames

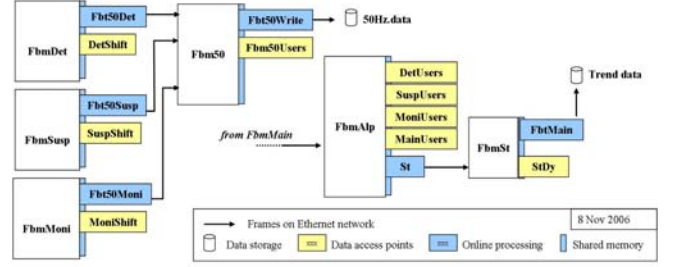


Fig. 5. Online processing tasks (blue color) and online data access points (yellow color) within the DAQ architecture, in November 2006.

collectors and provides resampled data (after anti-aliasing filter of 7th order) written on disk over several years. This allows a fast access to several weeks of data in order to follow and study the low frequency behaviour of the interferometer. The trend data are statistics (mean, rms, max, min) computed for each channel by a dedicated process (FbtMain on Fig.5) and written on disk along with the slow monitoring data. The trend data stream allows to monitor over several months the evolution of the detector, of the controls, of the environment or of the DAQ itself.

The access to those various data streams can be done online thanks to access points (or frames providers) connected to the shared memory of one of the frames collectors. Those access points (see Fig.5) are based on the FdIOServer described in section II-C. They have no link with the DAQ except the shared memory where they read the data. They consume at most some percents of CPU and allow to connect online, through Ethernet, any tool developed in Virgo for the visualisation, monitoring or writing of the data.

IV. DAQ USAGE IN INTERFEROMETER'S CONTROLS

A. DAQ and interferometer's digital control loops

As already mentioned, several signals acquired by the DAQ, including the dark fringe signal, are also used in the controls of the interferometer. The speed and stability of those controls put strong constraints on the DAQ readouts that must be fast and reliable enough. The main control loops tightly linked to the DAQ readouts are the following:

- **The longitudinal global control:** it uses photodiodes signals read by the DAQ at 10 kHz to compute the correction signals that drive the mirrors longitudinal position and control thus the length of the different optical cavities of the interferometer.
- **The alignment global control:** it uses quadrant photodiodes signals read by the DAQ at 500 Hz to determine the beams position and to compute the correction signals that drive the mirrors angular position.
- **Each suspension local control and inertial damping:** it uses the camera signals read by the DAQ at 50 Hz to compute the correction signals that drive the suspension's position and orientation.

- **The laser frequency and input mode-cleaner length control:** it uses photodiodes signals read by the DAQ at 10 kHz to compute the correction signals that drive the input laser beam frequency and the input mode-cleaner resonance length.

B. DAQ and interferometer's startup automation

Until 2005, to put the interferometer in a given configuration (for instance Fabry-Perot cavities aligned and locked, and interferometer locked on the dark fringe), a sequence of monitoring and manual actions was required from operators in control room. It appeared rapidly that, knowing the signals to monitor through the DAQ and using numerical controls, this interferometer's startup sequence could be automatized. A software named "Alp" has been developed in Virgo for this automation. Practically, a set of Alp processes gets data from the shared memory of a dedicated DAQ frames collector (FbmMain) and, by successive steps, controls the lock of the interferometer until its working point. Those steps are numbered from 1 to 12, among which the main ones are:

- **Step 1:** The mirrors are pre-aligned (except recycling mirror), the Fabry-Perot cavities are locked and the interference is locked on "gray fringe".
- **Step 4:** The recycling mirror is aligned and the recycling cavity is locked. The light power on the beam splitter is then of several hundreds of watts.
- **Step 8:** The interference is locked on dark fringe and the global control of alignment is activated.
- **Step 10:** The output mode-cleaner is locked on the laser TEM00 mode and the dark fringe control uses photodiode B1 placed after output mode-cleaner.
- **Step 12:** The tidal control is activated as well as the reallocation of controls on intermediate part of mirrors suspensions.

Then, reduction of control noises is activated, permanent calibration signals are injected on mirrors longitudinal controls and "Science Mode" flag is switched on. Each step of this locking sequence is defined in the configuration of an Alp process, which contains a set of conditions on the signals provided by the DAQ and a set of actions to be done on the interferometer. Those actions are done through Ethernet by sending Cm messages [17] to the softwares that control the interferometer's sub-systems. Each step defined in the configuration can be conditioned by the result of the previous steps and the full sequence of controls can thus be done easily and quickly. The low latency of the DAQ (about 2 s) allows the Alp automation software to have enough control speed and to complete the whole automatized sequence in less than 15 mn, if we exclude the current problem of mirrors thermalization at step 8 (see Fig.6).

V. DAQ MONITORING AND PERFORMANCES

As we will see in section VI, the Virgo detector and its environment are permanently monitored, but this does not guarantee the availability and the validity of the data. At each level of the data acquisition chain (front-end, frame builders,

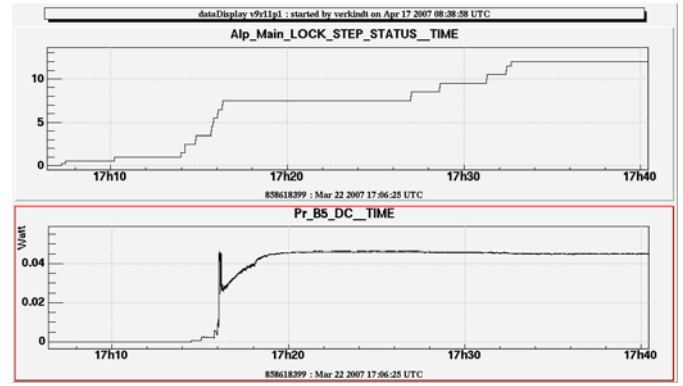


Fig. 6. A typical sequence of the automatized interferometer startup procedure. Upper plot shows the evolution of the `Alp_Main_LOCK_STEP_STATUS` value that monitor the startup steps. Lower plot shows the light power on the beam splitter that increases a lot when the recycling cavity is locked.

frames collectors), a failure can be at the origin of missing or corrupted data. Thus, all the processes involved in the data acquisition or in the data storage are monitored.

Using the GUI tool Xforms [21], a graphical interface [22] has been developed to control and monitor each DAQ process. This interface allows to start and stop the processes and to change their configuration. In addition, it provides several informations like memory consumption, process id, TCP name, etc... and builds a web page showing the online status of the DAQ processes with a latency less than 2 seconds. Fig.7 shows parts of the graphical interface used to monitor all the processes used in the data collection.

Collect	active	(golden) GP5 860925740 - Apr 18, 2007 (108) 12:02:07
TiServer	Golden	fr:00269 nm05350 GP5:860925740-2700 - Qc 0x0000e022
Detection DAQ line		
FfbDet	Golden	fr:00268 (1.85e+03 KB) [0/4] - sTime 0.78s
FfbDetP	Active	fr:00268 (1.14e+03 KB) [0/4] - lostGcAll - sTime 0.568s
FfbDet	Golden	fr:00268 (672 KB) [0/4] - sTime 0.291s
FfbDet	Golden	fr:00268 (972 KB) [0/4] - sTime 0.402s
FfbDet	Active	fr:00269 (18.067 KB) GP5860925740 Latency: 0.0787 - lostGc_Locking_mirror
FfbDet	golden	fr:00268 (4.67 MB) latency: 1.88 adc:495 sms:23

Fig. 7. Part of the graphical interface used to control and monitor the data collection DAQ processes. First column (colored buttons) show the status of the process. Second column (with arrows) give access to the logfile of each process. Detailed information (frame number, rate, latency,...) is provided on the next two columns.

In addition, some slow monitoring data are created by the data acquisition itself. Those data allow to monitor for instance the correctness of the timing stamp, the efficiency of the data acquisition, the total number of channels acquired, the data rate along the DAQ chain, the latencies introduced in the data acquisition chain. Among those data, the most important ones are visualized by trend plots on the web [23].

Since January 2002, those monitoring tools have allowed to estimate the performances of the data acquisition system in terms of stability, efficiency, data rate and latency. The main results are the following:

- The mean DAQ latency is less than 2 seconds at the level

of FbmMain. This 2 seconds latency and the reliability of the DAQ has allowed to develop the automation of the interferometer's locking steps through the DAQ.

- The data rate permanently sustained by the DAQ is about 20 MBytes/s (4 times larger than foreseen in the DAQ design), with more than 1350 fast channels (sampling frequency above 1 Hz) recorded.
- The percentage of data loss is difficult to estimate because the number of channels acquired fluctuates according to the commissioning needs. Nevertheless, it has been estimated on the dark fringe signal (main signal that will be used for gravitational waves search). It is less than 0.1% for each commissioning run and about 0.3% over the whole period from January 2004 to November 2006. This estimation takes into account all the periods of detector's maintenance, disks problems, network failure, upgrade of machines or operating system, stops of the DAQ (for architecture's modifications or for the use of new versions of the libraries).

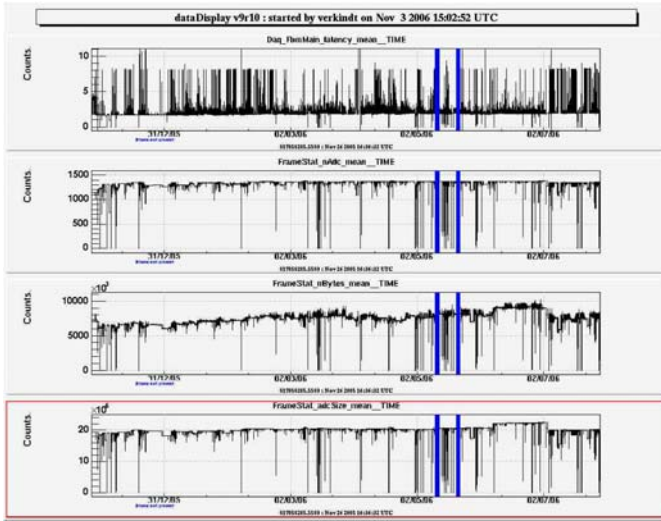


Fig. 8. From top to bottom: DAQ latency, number of fast channels and data rate (compressed and uncompressed) seen by DAQ between 26/11/2005 and 31/07/2006.

The two lower plots of Fig.8 show, over about one year, the continuous increase of the data rate (compressed and uncompressed), while the two upper plots show the relative stability of the mean latency and the increase of the number of fast channels collected. Table I shows the evolution of some DAQ numbers since July 2002, where we can see the increase of readout processes, channels number and data rate, while the latency and data loss are reduced down to 2 seconds and 0.3% respectively. Table II shows the duty cycle of the data acquisition between January 2004 and September 2006 and over most of the commissioning runs. This performance is illustrated by Fig.9, which shows the integrated DAQ duty cycle over one year.

The Virgo data acquisition system has run permanently

since early 2001, with the good performances described above. It sustained an increasing number of channels reaching now more than 1350 "fast" signals. It corresponds to a raw data rate of 20 MBytes/s currently provided to control room with a mean latency of 2 seconds.

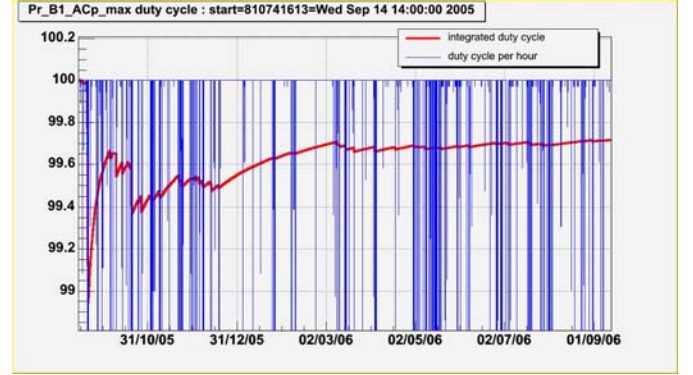


Fig. 9. DAQ duty cycle estimated by the presence of the dark fringe channel in the stored raw data, between sept. 2005 and sept. 2006.

TABLE I

EVOLUTION OF SOME PARAMETERS OF DAQ OVER THE LAST 4 YEARS. THE FULL LATENCY IS TAKEN BETWEEN FRONT-END READOUT AND DATA STORAGE (IN 2006, IT INCLUDED ABOUT 20 ADDITIONAL SECONDS DUE TO QUALIFICATION OF THE H RECONSTRUCTION PROCESS). THE ONLINE LATENCY IS TAKEN BETWEEN FRONT-END READOUT AND THE LAST FRAMES COLLECTOR (FbmMain) WHERE DATA ARE READ BY THE AUTOMATION PROCESSES. THE PERCENTAGE OF DATA LOST IS THE PERCENTAGE OF TIME WHERE THE DARK FRINGE SIGNAL WAS ABSENT.

bf Date	Jul2002	Jul2003	Feb2004	Dec2004	Aug2005	Nov2006
bf Run	E4	C0	C2	C5	C6	WSR5
bf Fast raw data channels	690	790	1180	1340	1330	1385
bf Trend data channels	3260	4000	5670	7880	7450	9473
bf Compr. data rate (MB/s)	4.2	4.7	5.8	6.7	6.3	7.9
bf Front-end readouts	24+24	—	31+45	31+78	29+65	30+72
bf Frame builders	22	—	30	30	31	34
bf Frames collectors	4	4	5	7	8	11
bf Data storage (TB)	2	6	32	70	120	120
bf Full latency (s)	8	8	7	6	4	22.7
bf Online latency (s)	8	8	6	5	3	1.9
bf Data lost percentage	—	—	—	0.18%	0.24%	0.023%
bf Data streams	2	3	4	6	6	6

VI. ONLINE DETECTOR MONITORING

Virgo must be operated 24h per day while very exposed to instrumental and environmental noises because of its high sensitivity. This constraint requires a highly selective data analysis and a reliable data acquisition. But it requires also a permanent monitoring of the interferometer's running conditions, first to provide in control room a quickly updated information on the detector's behaviour, then to help in the selection of data segments where data analysis can be done with confidence. Virgo has developed since several years some software tools for the online monitoring of the detector. Most of those tools are presented here. They use often the software libraries created for the DAQ and, thus, benefit from their reliability.

TABLE II

ESTIMATION OF THE DAQ DUTY CYCLE BETWEEN JANUARY 2004 AND SEPTEMBER 2006 FOR THE TREND DATA STREAM AND THE RAW DATA STREAM (DARK FRINGE SIGNAL ONLY). ALSO SHOWN IS THE DAQ DUTY CYCLE FOR EACH COMMISSIONING RUN SINCE JANUARY 2004.

bf Data stream	bf Epoch	bf Start GPS	bf duration	bf Duty cycle
Trend data	Jan2004-Jan2005	757585800	31400000	96.509%
Trend data	Sep2005-Sep2006	810741613	31400000	97.770%
Raw data	Jan2004-Jan2005	757585800	31400000	99.997%
Raw data	Sep2005-Sep2006	810741613	31400000	99.716%
Raw data	C5 (02 Dec 2004)	786031360	436000	99.828%
Raw data	C6 (28 Jul 2005)	806602213	1296000	99.760%
Raw data	C7 (14 Sep 2005)	810741613	403200	99.987%
Raw data	WSR1 (08 Sep 2006)	841784413	201600	100.000%
Raw data	WSR2 (23 Sep 2006)	843001214	201600	99.987%
Raw data	WSR5 (10 Nov 2006)	847231214	201600	99.977%
Raw data	WSR6 (01 Dec 2006)	849045614	201600	99.995%
Raw data	WSR7 (12 Jan 2007)	852678014	201600	99.940%
Raw data	WSR8 (9 Feb 2007)	855093614	201600	99.976%

A. The data display

A first mandatory need of the Virgo commissioning was to monitor online from the control room a large number of signals provided by the interferometer. To this purpose, a software tool dedicated to the visualization of the data has been developed for the Virgo experiment: the dataDisplay [24]. It allows, through a "buttons and browsers" graphical interface, to read, online or offline, the frame formatted data produced in Virgo. Fig.10 shows one of the dataDisplay's graphical interface panels.

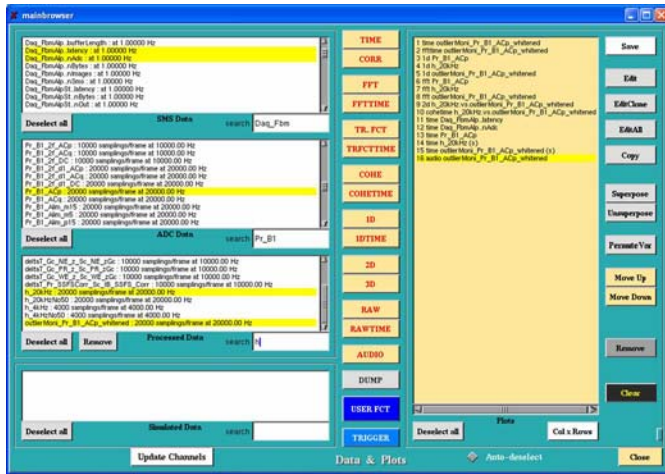


Fig. 10. Graphical panel to manage the signals and the plots of the dataDisplay. It allows to visualize the signals available in each frame, to choose some of them and the type of plots to display, then to manage the superposition of the plots or to edit the parameters of each of them.

Over the last years, the dataDisplay has enriched with several functionalities and new types of plots. It is able to create and edit plots that could help for the monitoring of the detector (time plots, spectra, 1D or 2D distributions, beams imaging...) or the signals analysis (transfer function, coherence plot,

spectrograms...). The dataDisplay can also do operations on the signals in order to monitor some combinations of signals used in the controls. It can superpose plots (including plots of data taken at different times), do operations on the plots, view the content of the frames or listen to the data. Fig.11 shows an example of online data visualization, during the WSR1 run in september 2006.

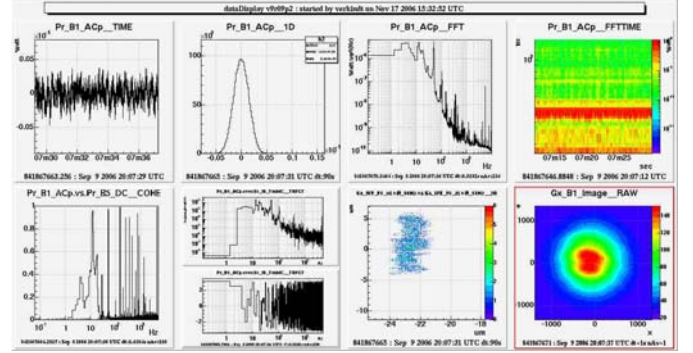


Fig. 11. Examples of plots created online by the dataDisplay with the data received from the DAQ. On first row are visible, from left to right, the dark fringe signal versus time, its distribution, its frequency spectrum and its time-frequency spectrogram. On second row are shown the coherence and the transfer function (module and phase) between the dark fringe signal and the light power on beam splitter. Last two plots monitor the horizontal position of the upper part of west mirror suspension and the profile of the beam at the output of the interferometer.

The dataDisplay uses open source code: a graphical interface created with the Xforms library [21], a data transfers management based on the frames library [13], the Fd and Cm libraries, and a plots visualization based on the Root library developed at CERN [25]. It has been compiled and used under OSF1 v5, Mac OSX, SunOS, Linux RedHat 7.2, Linux RedHat 9 or Scientific Linux 4.2.

The dataDisplay has been widely used in the Virgo control room to monitor online the detector. It has evolved over several years, according to the needs of the Virgo commissioning, and is still improving, focalizing now on the data analysis needs.

B. Monitoring through the web

In parallel to the online visualization provided by the dataDisplay, a set of web pages has been developed to provide over a longer time scale (some hours), but with a larger latency (half an hour), an overview of the state of the interferometer [23]. Those web pages are generated thanks to the "trend data" stream and thanks to a software environment developed for Virgo data analysis [26], VEGA, which mainly adapts to the Virgo needs the Root library developed at CERN. Thanks to the C/C++ interpreter of Root, a set of macros has been developed to monitor the long term evolution of the detector's subsystems (mirrors alignment, light power inside the recycling cavity, current sensitivity, DAQ data rate, etc...). Such macros are also used now to monitor the results of glitches search algorithms, to follow the amplitude and frequency of spectral lines, to follow the evolution of the parameters used in

the calibration, the level of noise in various frequency bands, the spectrograms of several useful signals, etc...

C. Online data qualification

In addition to the data visualization through dataDisplay or web monitoring, a mandatory need of the Virgo experiment was to set discriminant quantities (quality flags) that summarize the state of the detector. An architecture that uses the DAQ tools has been setup to generate those quality flags, record them in the Virgo data stream and visualize them in control room.

For each Virgo sub-system (suspensions, photodiodes, laser injection, environment monitoring, etc...), a set of quality flags is produced online by a dedicated algorithm. Each algorithm is made of a set of conditions applied on signals. From this set of flags a global quality flag is created. Then, all the flags are displayed online through web pages [27] to inform control room operators about any problem detected in the interferometer or its environment. In parallel, all the flags are also sent to the DAQ and stored on disk to help in the selection of data segments where the interferometer is in good conditions for data analysis.

Technically, all the detector monitoring algorithms are connected to the shared memory of a specific frames collector and use the same software implementation, the same libraries and the same type of configuration syntax. This is done thanks to a library (MoniL) developed to generate the algorithms and manage the quality flags through a configuration file, where the qualification of the data can be modulated according to the steps of the interferometer's startup procedure.

VII. CONCLUSION

Since summer 2001, the data acquisition of Virgo has run almost continuously, with regular improvements but without changing the main architecture, the software libraries or the standard tools chosen (VME, Ethernet, optical links, LynxOS, Linux, shared memory...). It has fulfilled its requirements for data loss minimization, latency reduction or data rate management. The software tools developed for the data collection have provided a good flexibility that has allowed for instance to connect easily new online processings or online monitoring tools.

Research and development is currently done to improve the Virgo detector in the coming years. This improved detector will have faster control loops with reduced noise levels. In this perspective, an upgrade of the VME boards used for the timing distribution and for the signals optical propagation is currently under way [28]. This new "front-end" acquisition will offer more possibilities and will handle the faster control loops, at least doubling the data rate to collect. On the contrary, thanks to the choice we made of standard tools like Ethernet network and shared memory, no new heavy development is foreseen for data collection or online processing which may

adapt (using faster and more numerous Linux workstations) to the new data rate and to possible new online processing requirements.

REFERENCES

- [1] *The Virgo Collaboration Final Design Report*, 1997, Virgo technical note VIR-TRE-DIR-1000-13
- [2] *The status of Virgo*, F. Acernese et al., PoS(HEP2005)029.
- [3] *Data Acquisition and Online Processing for the VIRGO experiment*, Proceedings of IXth Marcel Grossmann Meeting on General Relativity, Roma (2-8 Juillet 2000)
- [4] *Virgo Data Acquisition*, F. Acernese et al., Proceedings of the IEEE NPSS Real Time conference, Montreal, May 2003.
See also <http://lapp.in2p3.fr/virgo/daqManual/VirgoDAQ-RT2003.pdf>
- [5] *Sensing and control in Virgo Experiment*, Barsotti L., Nuclear Science Symposium Conference Record, 2004 IEEE Volume 7, 16-22 October 2004, 4625-4628
- [6] *The real-time distributed control of the Virgo Interferometric Detector of Gravitational Waves*, F. Acernese et al., Proceedings of the IEEE NPSS Real Time Conference, Chicago, Mai 2007
- [7] *RIO 8062: PowerPC based RISC I/O boards*, (400 MHz, 32 or 64 MBytes RAM), DOC 8062 UM CES, 1998, Creative Electronics Systems S.A., 38 av. Eugène-Lance PO Box 584 CH-1212 Grand-Lancy 1, Switzerland
- [8] *The Virgo Timing system*, Virgo note VIR-TRE-LAP-5200-103, 1997
- [9] *The Digital Optical Link boards*, Virgo note VIR-SPE-LAPP-5200-105, 2000
- [10] *The Digital Camera VME readout*, F. Bellachia et al., Nucl. Instrum. Methods A 413 (1998), 151
- [11] *Timing performances*, Internal notes LIGO-T020036-00-D and VIR-NOT-LAP-1390-198, 2002
- [12] *Frame format specifications*, Internal notes LIGO-T970130-F-E and VIR-SPE-LAP-5400-102
See also <http://lapp.in2p3.fr/virgo/FrameL>
- [13] *Frame Library*, Virgo note VIR-MAN-LAP-5400-103, 1997
See also <http://lapp.in2p3.fr/virgo/FrameL>
- [14] *Frame Vector Library*, see <http://lapp.in2p3.fr/virgo/FrameL>
- [15] *Frames distribution on Ethernet*, Virgo note VIR-SPE-LAP-5400-105, 2004
- [16] *Frame distribution through shared memory*, Virgo note VIR-MAN-LAP-5400-113, 2000
- [17] *The Cm library*, Virgo note VIR-MAN-LAL-5100-112, 1998
- [18] *The Slow Frame Builder*, Virgo note VIR-MAN-LAP-5400-109, 2000
- [19] *The Main Frame Builder*, Virgo note VIR-MAN-LAP-5400-112, 2000
- [20] *Offline storage system*, Virgo note VIR-TRE-CAS-7100-103, 2000
- [21] *The Xforms Graphical User Interface library*, T.C. Zhao, A. Leeming <http://savannah.nongnu.org/projects/xforms>
- [22] *Cl Graphical Interface*, Virgo note VIR-MAN-LAP-5400-119
See also <http://wwwcascina.virgo.infn.it/sDoc/virgoDev/Cl/v3r12/doc/ClDoc.doc>
- [23] See <http://wwwcascina.virgo.infn.it/MonitoringWeb/General>
- [24] *Data Display*, Virgo note VIR-SPE-LAP-5400-108, 2004
See also <http://lapp.in2p3.fr/virgo/dataDisplay>
- [25] *Root library, for data handling and visualization*, <http://root.cern.ch>
- [26] *Vega, an environment for gravitational waves data analysis*, Int. Journal of Modern Physics D9 Issue 3 (2000) 293
See also <http://lapp.in2p3.fr/virgo/vega>
- [27] See <http://olserver13.virgo.infn.it/itf/qcmoni/V5/index.php>
- [28] *Control System Upgrade of the VIRGO interferometer. Timing Interface & Optical Links*, Virgo Note VIR-NOT-LAP-1390-278